SR-IDS: A Novel Network Intrusion Detection System Based on Self-taught Learning and Representation Learning

Qinghao Wang, Geying Yang, Lina Wang(✉), Jie Fu, and Xiaowen Liu

School of Cyber Science and Engineering, Wuhan University, Wuhan, China
{anthony,yanggeying,lnwang,whuerfu,2021202210085}@whu.edu.cn

Abstract. As a proactive network security protection scheme, network intrusion detection system (NIDS) has become a powerful tool for early warning of computer and communication systems attacks. However, traditional machine learning methods struggle to pay attention to both spatial and temporal features of network traffic simultaneously, resulting in poor detection performance. In this paper, we propose SR-IDS, an Intrusion Detection System based on Self-taught learning and Representation learning, which consists of one-dimensional stacked convolutional autoencoders (1D-SCAE) and bidirectional gated recurrent units (BiGRU). SR-IDS can extract spatial features through 1D-SCAE and abstract temporal features via BiGRU. It uses self-taught learning and representation learning to simultaneously focus on the spatial and temporal characteristics of network traffic, overcoming the challenges of traditional methods in feature extraction. Experiments show that our SR-IDS model can distinguish the network traffic with 98.90% accuracy on the UNSW-NB15 dataset.

Keywords: Convolutional autoencoders · Feature extraction · Recurrent neural networks · Traffic intrusion detection

1 Introduction

Network intrusion detection system (NIDS) monitors all traffic in the network and detects each data packet passing through the web. Many researchers have begun studying intrusion detection techniques to deal with network attacks effectively. In classification problems, machine learning algorithms perform feature extraction to identify malicious behaviors in network traffic [8]. However, the statistical characteristics of traffic have changed considerably in terms of network architectures and applications today. Traditional machine learning methods have been powerless to efficiently and accurately abstract spatial and temporal features of abnormal traffic.

Self-taught learning is a typical machine learning framework for using unlabeled data in supervised classification tasks [22]. The method does not require the assumption that unlabeled data follows the same distribution as labeled data.

L. Iliadis et al. (Eds.): ICANN 2023, LNCS 14256, pp. 554–565, 2023.
https://doi.org/10.1007/978-3-031-44213-1_46

Besides, representation learning analyzes the characteristic of data that makes it easier to extract helpful information when building predictors [5]. Inspired by the above ideas, we develop a noval network intrusion detection system based on self-taught learning and representation learning.

General traffic features can be divided into two categories: spatial features, such as data packet features, and temporal features, such as network flow features. NIDS often struggles to broaden the horizon and jump out of the local optimum solution when using only spatial or temporal features [27]. In this paper, we design one-dimensional stacked convolutional autoencoders (1D-SCAE), an excellent self-taught learning model which abstracts spatial features by reducing the dimensionality of complex data signals. Besides, bidirectional gated recurrent units (BiGRU) can extract temporal features of traffic sequences in representation learning. Therefore, we propose a deep neural network model based on 1D-SCAE and BiGRU, which can accurately extract spatial and temporal features and enhance the performance of malicious traffic detection. The main contributions of the proposed work include the following:

- We design 1D-SCAE—an improved network traffic spatial feature extraction model, which uses sparse regularization to reduce overfitting by invalidating a certain part of active neurons. The greedy layer-wise strategy is adopted to achieve the best detection performance.
- We propose a BiGRU-based temporal feature extraction model that utilizes TimeseriesGenerator to generate and model traffic time series. It can acquire both memories from history and information from the future.
- We develop SR-IDS, a network intrusion detection system that simultaneously focuses on network traffic's spatial and temporal characteristics. Experiments show that the accuracy of SR-IDS on the UNSW-NB15 dataset can reach 98.90%.
- We discuss different hyperparameters to determine the optimal model architecture. Furthermore, we compare the detection performance of different RNN variants.

The rest of the paper is organized as follows. The related work on NIDS is reviewed in Sect. 2. Then we present the details of the proposed SR-IDS in Sect. 3. The accuracy and the efficiency of SR-IDS are verified in Sect. 4 by comparing it with several state-of-the-art IDS algorithms. Finally, we provide our conclusions and discuss the future work in Sect. 5.

2 Related Work

NIDS is a necessary foundation and premise for dealing with complex network attacks and identifying malicious traffic behavior. The deep learning models currently applied to network anomaly detection include two categories: generative intrusion detection model and discriminative intrusion detection model.

2.1 Generative Intrusion Detection Model

Generative models often adopt an advanced hierarchical learning method to establish a multi-level model, which can flexibly analyze and restore joint probability distribution. The current famous generative model architecture mainly includes autoencoder and its variants [18].

Amir et al. [4] designed a new lightweight architecture that considers feature separation and uses surrounding information of a single value in the feature vector. The accuracy is improved while reducing the memory footprint and the need for processing power. Iliyasu et al. [12] achieved a few-shot learning intrusion detection, which uses the feature extraction model in the few-shot learning stage to fit a classifier with a small number of novel attack samples. Long et al. [17] proposed a network intrusion detection model based on an integrated autoencoder. It uses recursive feature addition to select the optimal subset of features, which can significantly reduce the training time and improve the intrusion detection performance.

2.2 Discriminative Intrusion Detection Model

Discriminative models are usually based on the excellent classification of heterogeneous data to achieve the best recognition. The common discriminative model structures mainly include recurrent neural networks and convolutional neural networks [2].

Imrana et al. [14] proposed a novel feature-driven intrusion detection system. The model first utilizes a statistical model to rank all the features, then uses best-first-search algorithm to search for the best subset, and finally classifies testing data based on the best subset. Sahu et al. [23] proposed a multi-classification intrusion detection method based on LSTM and fully connected networks. This method accurately classifies the imbalanced intrusion data. Imrana et al. [13] used an improved RNN model for network intrusion detection, which can be associated with the feature knowledge and accurately classify unknown data.

Several works sought to propose ML-based solutions with consideration of as many essential features as possible, and the approaches managed to obtain interesting results. However, there are still some challenges in extracting both spatial and temporal traffic features. Inspired by existing research progress, we propose SR-IDS—a new intrusion detection system with the advantages of generative models and discriminative models. Moreover, it can serially extract the spatial and temporal features of network traffic accurately.

3 The Proposed Model

In this section, we introduce how SR-IDS works. SR-IDS first preprocesses the UNSW-NB15 dataset, including one-hot encoding and normalization. Afterward, SR-IDS uses 1D-SCAE to extract spatial features of network traffic, and the greedy layer-wise strategy is adopted to pre-train the neural network. Finally,

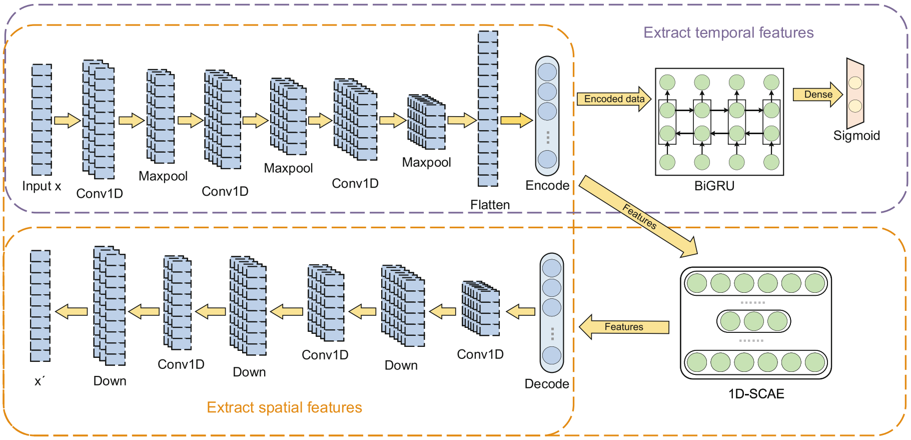

Fig. 1. The framework of our proposed SR-IDS model. 1D-SCAE (marked in blue) extracts spatial features through encoding and decoding. The output of the last encoding layer of 1D-SCAE is the input of BiGRU (marked in green). BiGRU extracts temporal features by generating time series. (Color figure online)

SR-IDS uses BiGRU to extract temporal features of network traffic. BiGRU accepts input from pre-trained 1D-SCAE and outputs to the binary classifier. Figure 1 describes the framework of our proposed SR-IDS model.

3.1 Data Preprocessing

In general, machine learning models can only process meaningful numerical data, but the actual data differs from what we expected. In order to enable machine learning models to process and analyze traffic data, assigning numerical meaning to features is necessary. One-hot encoding is a commonly used feature encoding method.

One-hot encoding expresses a specific type of different values in binary vectors. The N values used for encoding correspond to the states of N registers one by one. Only one bit in any form is activated, and the rest of the registers are inactive. The specific representation is generally $v_i = \{0, 1, 0, \ldots 0, 0\}$, and the dimension of the vector is equal to the number of possible values N of the eigenvalues to be encoded.

After encoding, we use the min-max method to standardize network traffic samples. With a fixed output range, the min-max method performs a linear operation on the sequence $\{x_1, x_2, \ldots, x_n\}$. After transformation, the new sequence $\{y_1, y_2, \ldots, y_n\} \in (0, 1)$ are dimensionless:

$$y_i = \frac{x_i - \min_{1 \leq i \leq n}\{x_j\}}{\max_{1 \leq i \leq n}\{x_j\} - \min_{1 \leq i \leq n}\{x_j\}} \tag{1}$$

It can be found that min-max forces the original input data to distribute in [0, 1], and the normalized scale transformation is only related to extreme values.

3.2 Spatial Feature Extraction

Spatial features of network traffic refer to feature sets related to packets, for example, packet size and number. We design a 1D-SCAE for spontaneously learning spatial feature representation, and Fig. 2 describes the architecture. In each layer, the autoencoder convolves the features of the lower layers to produce a high-level representation. The whole methodology is shown as follows:

$$x_j^l = f(\sum_{i \in M_j} x_i^{l-1} \times k_{ij}^l + b_j^l) \tag{2}$$

where M_j represents the input feature map, l represents the l-th layer in 1D-SCAE, and k is the convolution kernel. f represents the activation function, and b_j^l is the bias vector.

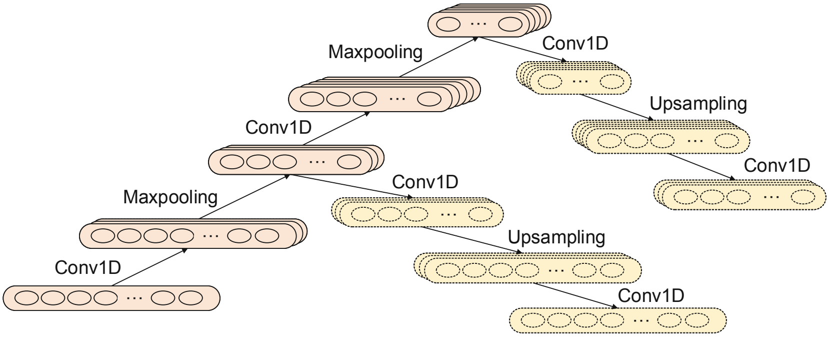

Fig. 2. The structure of proposed 1D-SCAE model

The 1D-SCAE consists of three convolutional autoencoders, and their encoder layers are stacked in the model construction process to build the complete 1D-SCAE model. After the training is completed, we discard the decoders and connect the last encoder layer to the subsequent temporal extraction model, which will be explained in the next subsection. MSE Loss is used to evaluate the effect of feature extraction and input reconstruction as follows:

$$J = \frac{1}{n} \sum_{i=1}^{n} (x_i - x_i')^2 \tag{3}$$

where i is the sample index, x_i is the original input data, and x_i' is the reconstructed data after dimensionality reduction by 1D-SCAE.

We also add a custom regularization term in 1D-SCAE to improve the generalization performance of the model. The principle is that different inputs cause different neurons to be activated, making neurons better dependent on data. In general, the constant ρ is the proportion of activated neurons, which is used to measure the average activity $\hat{\rho}$ of the activation degree of neurons:

$$\hat{\rho} = \frac{1}{N}\sum_{i=1}^{N}\Theta(x_i) \tag{4}$$

where N is the number of neurons in the hidden layer, Θ is the corresponding neuron transformation. In the field of machine learning, forward KL divergence is often used as the training cost to measure the difference between two probability distributions. Forward KL divergence makes sure $\hat{\rho}$ close to ρ, and the regularization term punishs the deviation between $\hat{\rho}$ and ρ:

$$KL(\rho\|\hat{\rho}) = \rho\log\frac{\rho}{\hat{\rho}} + (1-\rho)\log\frac{1-\rho}{1-\hat{\rho}} \tag{5}$$

If $\hat{\rho}$ is equal to ρ, the KL divergence is 0; otherwise, it will gradually increase as the difference between ρ and $\hat{\rho}$ increases. Therefore, the error function $J^{'}$ in the sparse autoencoder is shown as follows:

$$J^{'} = J + \mu\sum_{j=1}^{N}KL(\rho\|\hat{\rho}) \tag{6}$$

where J is the error when no sparse item is added, and μ is the impact factor used to balance the weight of KL divergence in the entire loss function.

3.3 Temporal Feature Extraction

In this work, we group traffic records by timestep and link the context with their labels. Our proposed SR-IDS can accurately reflect the time characteristics of network traffic and significantly reduce the false positive rate.

SR-IDS takes the output from the spatial feature extraction model as input and uses TimeseriesGenerator—a time series generator to convert isolated samples into a sequence. After serialization, the processed traffic is input into the BiGRU. The principle of BiGRU is to split the neurons of a regular GRU into two directions, one for positive time direction and another for negative time direction.

Assume that the current input vector is x_t, the last step activation vector is r_{t-1}, W and U are weight matrices used to represent the connection strength between neurons, and b is the bias vector. σ_g represents the sigmoid activation function, the update gate vector z_t and the reset gate vector r_t are shown as follows:

$$\begin{aligned} z_t &= \sigma_g(W_z x_t + U_z h_{t-1} + b_z) \\ r_t &= \sigma_g(W_r x_t + U_r h_{t-1} + b_r) \end{aligned} \tag{7}$$

The candidate activation vector $\hat{h}_t$ is obtained through the Hadamard product of r_t and h_{t-1}, where ϕ_h represents the hyperbolic tangent function:

$$\hat{h}_t = \phi_h(W_h x_t + U_h(r_t \odot h_{t-1}) + b_h) \tag{8}$$

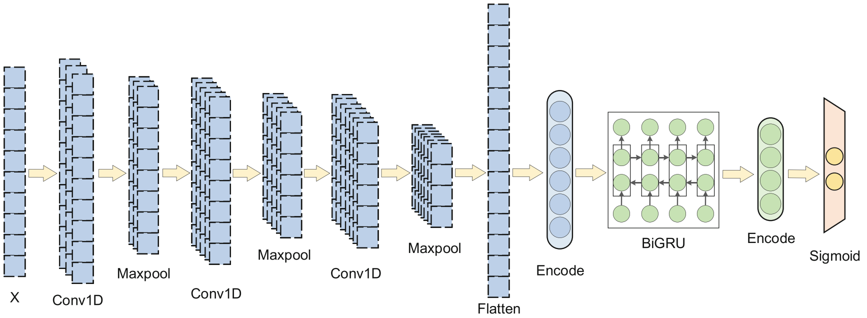

Fig. 3. Classification model based on 1D-SCAE and BiGRU

Finally, update the activation output vector of the hidden unit h_t at time t:

$$h_t = (1 - z_t) \odot h_{t-1} + z_t \odot \hat{h}_t \tag{9}$$

When the 1D-SCAE model is completely trained, we connect it with the subsequent BiGRU network, as shown in Fig. 3. We optimize the free parameters in BiGRU to achieve the global optimum. The binary cross entropy loss function in the final binary classification is adopted to evaluate the model as follows:

$$\xi = -\frac{1}{N}\sum_{i=1}^{N} y_i \log(p(y_i)) + (1 - y_i)\log(1 - p(y_i)) \tag{10}$$

where i is the sample index, N is the number of samples, y_i is the binary label of the i-th sample, and $p(y_i)$ is the probability that the output belongs to the y_i label. For the case where the label y_i is 1, if the predicted value $p(y_i)$ approaches 1, then the loss approaches 0. Conversely, if the predicted value $p(y_i)$ approaches 0, the loss should be tremendous.

4 Experiments

In this section, a series of experiments are conducted to verify the efficiency and accuracy of the proposed SR-IDS. Specifically, we first present the experimental settings and some details. Then we analyze some critical parameters to find the optimal solution. Lastly, we evaluate SR-IDS's performance and compare it with some state-of-the-art methods.

4.1 Dataset

The UNSW-NB15 dataset simulates a modern representation of network traffic [19]. Each instance in the dataset is a network flow that summarizes the activity of a sequence of unidirectional packets with contextual features. Additional features are introduced into the dataset, totaling 49 features.

Table 1. Model hierarchy and some parameters

Layers	Types	Input size	Output size
1	Conv1D	None, 200, 1	None, 200, 8
2	MaxPooling1D	None, 200, 8	None, 100, 8
3	Conv1D	None, 100, 8	None, 100, 16
4	MaxPooling1D	None, 100, 16	None, 50, 16
5	Conv1D	None, 50, 16	None, 50, 32
6	MaxPooling1D	None, 50, 32	None, 25, 32
7	Flatten	None, 25, 32	None, 800
8	Dense	None, 800	None, 32
9	TimeseriesGenerator	None, 32	None, 8, 32
10	BiGRU	None, 8, 32	None, 8, 48
11	BiGRU	None, 8, 48	None, 8, 24
12	Dense	None, 8, 24	None, 6
13	Dense	None, 6	None, 1

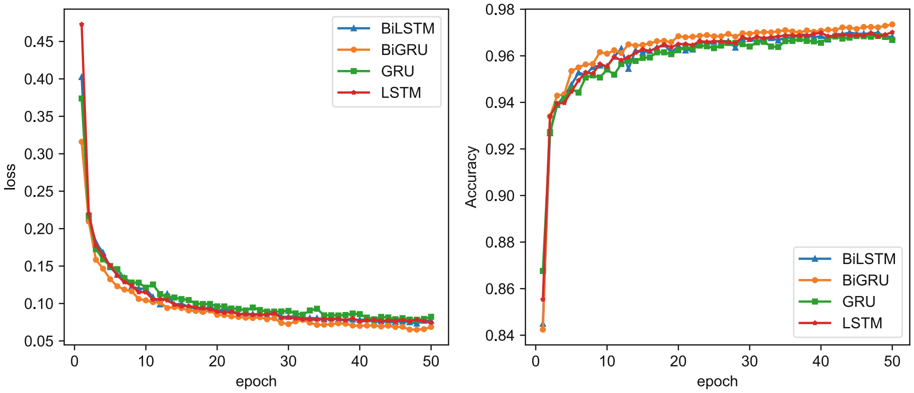

Fig. 4. Training loss and accuracy of different RNN variants

4.2 Model Hierarchy

SR-IDS inputs the preprocessed data into three independent one-dimensional convolutional autoencoders and trains them separately through a greedy layer-wise strategy. Three autoencoders' encoder layers are stacked after training by the weight-sharing method and then connects to the time series generator to produce traffic groups with contextual features. Afterwards, we use BiGRU to extract temporal feature and output the type judgment of the testing set. The complete model hierarchy and some significant parameters are shown in Table 1.

4.3 Parameters Analysis

We compare and test the influence of different learning rate and dropout ratio on convolutional layer and dense layer, as shown in Table 2. We find that the convergence speed of the entire neural network is extremely slow when the initial learning rate of the dense layer is less than 0.0001. It means the time overhead significantly increases, and the effect improvement is negligible, so we do not adopt the lower initial learning rate scheme.

Table 2. Comparison of different learning rate and dropout ratio

Layers	Learning rate	Dropout	Accuracy	FAR	F1 score
Conv1D	0.005	0.05	0.9310	0.0700	0.9184
		0.1	0.9358	0.0693	0.9300
		0.3	0.9187	0.0869	0.9226
	0.001	0.05	0.9398	0.0626	0.9301
		0.1	**0.9439**	**0.0537**	**0.9355**
		0.3	0.9106	0.0574	0.9280
	0.0005	0.05	0.9317	0.0604	0.9208
		0.1	0.9324	0.0586	0.9245
		0.3	0.9289	0.0654	0.9177
Dense	0.001	0.05	0.9401	0.0593	0.9353
		0.1	0.9439	0.0537	0.9355
		0.3	0.9422	0.0540	0.9320
	0.0005	0.05	0.9471	0.0569	0.9273
		0.1	0.9488	0.0525	0.9392
		0.3	0.9306	0.0528	0.9394
	0.0001	0.05	0.9533	0.0505	0.9314
		0.1	**0.9556**	**0.0499**	**0.9403**
		0.3	0.9485	0.0613	0.9345

We also compare the loss and accuracy of different RNN models during training iterations. Figure 4 shows the detailed performance of training loss and accuracy for attack detection. It can be seen that loss and accuracy hardly change when the epoch reaches 50, and BiGRU can achieve better performance than the other three RNN variants.

4.4 Evaluation

We compare the proposed method's performance with some state-of-the-art methods, as shown in Table 3. Additionally, we test our model on KDD CUP 99 [26] and CIC-IDS-2017 dataset [24], which also shows well performance. In summary, our proposed SR-IDS method can achieve excellent performance in network traffic anomaly detection.

Table 3. Comparison with other machine learning algorithms

Dataset	Model	Accuracy	Precision	Recall	F1 score
UNSW-NB15	DT [3]	88.30%	94.59%	77.78%	85.37%
	SVM [28]	89.63%	–	–	–
	GBT [30]	93.13%	92.38%	92.84%	92.61%
	GAN [7]	92.39%	91.46%	94.03%	94.39%
	CNN [6]	86.25%	86.92%	86.25%	86.59%
	Our SR-IDS	**98.90%**	**98.90%**	**98.90%**	**98.90%**
KDD CUP 99	DT [20]	94.46%	96.67%	–	–
	SVM [21]	96.61%	**98.04%**	95.31%	96.66%
	GBT [25]	91.82%	86.51%	–	–
	GAN [1]	–	86.76%	86.94%	85.71%
	CNN [29]	94.11%	–	93.22%	–
	Our SR-IDS	**98.15%**	97.26%	**99.01%**	**98.13%**
CIC-IDS-2017	DT [10]	94.48%	96.67%	–	–
	SVM [11]	93.75%	–	94.73%	–
	GBT [9]	**97.83%**	–	–	–
	GAN [16]	–	**98.17%**	90.57%	88.42%
	CNN [15]	97.39%	–	82.12%	–
	Our SR-IDS	96.16%	95.42%	**97.15%**	**96.28%**

5 Conclusion

In this paper, we propose SR-IDS, an intrusion detection system for network traffic based on self-taught learning and representation learning, which simultaneously focuses on traffic's spatial and temporal characteristics. Specifically, it utilizes 1D-SCAE to extract spatial features and BiGRU to extract temporal features. The greedy layer-wise strategy is adopted in the training process of 1D-SCAE, and sparse regularization is applied to reduce overfitting. BiGRU generates time series through TimeseriesGenerator to extract advanced time features. Multiple experiments have proved that BiGRU can achieve the best score among RNN variants. The accuracy rate of our proposed SR-IDS model in classifying network traffic on UNSW-NB15 dataset can reach 98.90%, which is more efficient than other current IDS methods.

In future research, we can consider online operations to improve robustness and stability. Furthermore, defense against attack techniques targeting deep learning models is also a research direction in the future.

Acknowledgements. This work is supported by the National Key Research and Development Program of China (No. 2020YFB1805403) and the National Natural Science Foundation of China (61876134).

References

1. Ahmad, R., Li, L.H., Sharma, A.K., Tanone, R.: Boundary-seeking GAN approach to improve classification of intrusion detection systems based on machine learning model. In: 2023 17th International Conference on Ubiquitous Information Management and Communication (IMCOM), pp. 1–5. IEEE (2023)
2. Alghanam, O.A., Almobaideen, W., Saadeh, M., Adwan, O.: An improved PIO feature selection algorithm for IoT network intrusion detection system based on ensemble learning. Expert Syst. Appl. **213**, 118745 (2023)
3. Anwer, H.M., Farouk, M., Abdel-Hamid, A.: A framework for efficient network anomaly intrusion detection with features selection. In: 2018 9th International Conference on Information and Communication Systems (ICICS), pp. 157–162. IEEE (2018)
4. Basati, A., Faghih, M.M.: PDAE: efficient network intrusion detection in IoT using parallel deep auto-encoders. Inf. Sci. **598**, 57–74 (2022)
5. Bengio, Y., Courville, A., Vincent, P.: Representation learning: a review and new perspectives. IEEE Trans. Pattern Anal. Mach. Intell. **35**(8), 1798–1828 (2013)
6. Cao, B., Li, C., Song, Y., Qin, Y., Chen, C.: Network intrusion detection model based on CNN and GRU. Appl. Sci. **12**(9), 4184 (2022)
7. Ding, H., Chen, L., Dong, L., Fu, Z., Cui, X.: Imbalanced data classification: a KNN and generative adversarial networks-based hybrid approach for intrusion detection. Futur. Gener. Comput. Syst. **131**, 240–254 (2022)
8. Du, R., Li, Y., Liang, X., Tian, J.: Support vector machine intrusion detection scheme based on cloud-fog collaboration. Mob. Netw. Appl. **27**(1), 431–440 (2022)
9. Faker, O., Dogdu, E.: Intrusion detection using big data and deep learning techniques. In: Proceedings of the 2019 ACM Southeast Conference, pp. 86–93 (2019)
10. Ferrag, M.A., Maglaras, L., Ahmim, A., Derdour, M., Janicke, H.: RDTIDS: rules and decision tree-based intrusion detection system for internet-of-things networks. Future Internet **12**(3), 44 (2020)
11. Gu, J., Lu, S.: An effective intrusion detection approach using SVM with naïve bayes feature embedding. Comput. Secur. **103**, 102158 (2021)
12. Iliyasu, A.S., Abdurrahman, U.A., Zheng, L.: Few-shot network intrusion detection using discriminative representation learning with supervised autoencoder. Appl. Sci. **12**(5), 2351 (2022)
13. Imrana, Y., Xiang, Y., Ali, L., Abdul-Rauf, Z.: A bidirectional LSTM deep learning approach for intrusion detection. Expert Syst. Appl. **185**, 115524 (2021)
14. Imrana, Y., et al.: χ 2-BidLSTM: a feature driven intrusion detection system based on χ 2 statistical model and bidirectional LSTM. Sensors **22**(5), 2018 (2022)
15. Le Jeune, L., Goedemé, T., Mentens, N.: Feature dimensionality in CNN acceleration for high-throughput network intrusion detection. In: 2022 32nd International Conference on Field-Programmable Logic and Applications (FPL), pp. 366–374. IEEE (2022)
16. Lee, J., Park, K.: AE-CGAN model based high performance network intrusion detection system. Appl. Sci. **9**(20), 4221 (2019)
17. Long, C., Xiao, J., Wei, J., Zhao, J., Wan, W., Du, G.: Autoencoder ensembles for network intrusion detection. In: 2022 24th International Conference on Advanced Communication Technology (ICACT), pp. 323–333. IEEE (2022)
18. Louk, M.H.L., Tama, B.A.: Dual-IDS: a bagging-based gradient boosting decision tree model for network anomaly intrusion detection system. Expert Syst. Appl. **213**, 119030 (2023)

19. Moustafa, N., Slay, J.: UNSW-NB15: a comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set). In: 2015 military communications and information systems conference (MilCIS), pp. 1–6. IEEE (2015)
20. Nancy, P., Muthurajkumar, S., Ganapathy, S., Santhosh Kumar, S., Selvi, M., Arputharaj, K.: Intrusion detection using dynamic feature selection and fuzzy temporal decision tree classification for wireless sensor networks. IET Commun. **14**(5), 888–895 (2020)
21. Nerlikar, P., Pandey, S., Sharma, S., Bagade, S.: Analysis of intrusion detection using machine learning techniques. Int. J. Comput. Netw. Commun. Secur. **8**(10), 84–93 (2020)
22. Raina, R., Battle, A., Lee, H., Packer, B., Ng, A.Y.: Self-taught learning: transfer learning from unlabeled data. In: Proceedings of the 24th International Conference on Machine Learning, pp. 759–766 (2007)
23. Sahu, S.K., Mohapatra, D.P., Rout, J.K., Sahoo, K.S., Pham, Q.V., Dao, N.N.: A LSTM-FCNN based multi-class intrusion detection using scalable framework. Comput. Electr. Eng. **99**, 107720 (2022)
24. Sharafaldin, I., Lashkari, A.H., Ghorbani, A.A.: Toward generating a new intrusion detection dataset and intrusion traffic characterization. ICISSp **1**, 108–116 (2018)
25. Tama, B.A., Rhee, K.H.: An in-depth experimental study of anomaly detection using gradient boosted machine. Neural Comput. Appl. **31**, 955–965 (2019)
26. Tavallaee, M., Bagheri, E., Lu, W., Ghorbani, A.A.: A detailed analysis of the KDD CUP 99 data set. In: 2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications, pp. 1–6. IEEE (2009)
27. Thakkar, A., Lohiya, R.: A survey on intrusion detection system: feature selection, model, performance measures, application perspective, challenges, and future research directions. Artif. Intell. Rev. **55**(1), 453–563 (2022)
28. Tian, Q., Li, J., Liu, H.: A method for guaranteeing wireless communication based on a combination of deep and shallow learning. IEEE Access **7**, 38688–38695 (2019)
29. Yong, L., Bo, Z.: An intrusion detection model based on multi-scale CNN. In: 2019 IEEE 3rd Information Technology, Networking, Electronic and Automation Control Conference (ITNEC), pp. 214–218. IEEE (2019)
30. Zhou, Y., Han, M., Liu, L., He, J.S., Wang, Y.: Deep learning approach for cyberattack detection. In: IEEE INFOCOM 2018-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), pp. 262–267. IEEE (2018)